DataAprori algorithm : Implementation of scalable Data Mining by using Aprori algorithm

M Afshar Alam[1] , Sapna Jain[2] ,Ranjit Biswas[3]
Department of Computer Science ,Jamia Hamdard
[1]mailtoafshar@rediffmail.com
[2] hellosap@sify.com
[3]ranjitbiswas@yahoo.com
NewDelhi,India-110062

Abstract **- Data Mining is discovery of a unknown relationships associations, groupings, classifiers from data. Association rule mining (ARM) is a knowledge discovery technique used in various data mining applications. The task of discovering scalable rules from the multidimensional database with reduced support is an area for exploration for research . Pruning is a technique for simplifying and hence generalising a decision tree. Error-Based Pruning replace sub-trees with leaves .It uses decision class is the majority. In this paper we have proposed an algorithm DataAprori to generate scaled rules using the alarm technique. Network problems manifest themselves as an alarm sequence. Since network problems repeat more or less frequently, processing of alarm sequences from alarm history can be good base for creation of correlation rules that will be used in the future, when the same problem will appear. In this paper we have proposed DataAprori that induces a set of rules of the potential usage of the mathematical Apriori algorithm in fault management introducing logical inventory data in typical alarm by introducing the sequence detection processes. Experimental on real world datasets show that the proposed approach improves performance over existing approach in the form of High level-correlations (alarm sequences) which are detected in a telecommunication network.**

Keywords: Data mining , ABCDE architecture ,pruning, Aprori technique.

I. Introduction:

AIREP learns the clauses in the order in which they will be used by a PROLOG interpreter. Before subsequent rules are learned, each clause is completed (learned *and* pruned) and all covered examples are removed. Therefore, the AIREP approach eliminates the problem of incompatibility between the separate-and conquer learning strategy and the reduced-error pruning strategy.Typically, a network problem is represented by the number of alarms coming from one or more network elements. If the alarms are coming from more than one network element, it is reasonable to expect that the network elements are interconnected. If we have a logical inventory database at our disposal (i.e., database where information about network element interconnections is stored), we can try to include it in the discovery environment. How? We can consider only the clusters containing alarms from interconnected network elements.

Since a logical inventory database is not always available, there is a possibility to “generate” it, based on the alarm historical data. In that case, we will first analyze alarms by their location only. After that analysis we will have information about the most frequent points of interconnection. This data can be stored in a logical inventory database (using a predefined threshold) and can be used in the cluster splitting process in the future. This concept is described in [7].

This research work is the extension of the previous work where we have proposed Aprori-UB which uses multidimensional access method UB-tree to generate efficient association rules with high support and confidence[19][20]. The Aprori-Ub approach reduces not only the number of item sets generated but also the overall execution time of the algorithm. In this paper we have used the abcde architecture for high-level correlations discovery as well as typical patterns that can be used for low-level correlations and filtrations[48][49].

The paper is organized as follows. Section 2 gives the overview of the previous work done in the same field. Section 3 explains the concepts used in this paper. Section 4 gives the proposed work. Section 5 gives the experimentation details. Section 6 and Section 7 discusses the conclusion and future scope.

II. Related Work

We define ,C_k as a candidate itemset of size k ,Z_k as a frequent itemset of size k, An AIREP algorithm is

1) Find frequent set L_{k-1}
2) Join step: C_k is generated by joining L_{k-1} with itself (cartesian product L_{k-1} x L_{k-1})
3) Prune step : Use theIncremental Reduced Error pruning to generate scalable single rule.
4) Frequent set L_k has been achieved.

The AIREP (Aprori Incremental Reduced Error Pruning) pseudo code :

```
AIREP (T, ụ)

Z1 ← large multidimensional itemsets that appear in more than

Of large item set ụ transactions

K ← 2

    While ( Z_{k-1} ≠ 0 )

    Ck ← Generate (Z_{k-1} )    // join and prune step

    // using IREP

    procedure I-REP (Examples, SplitRatio)

    Theory = ⍰ ;

While Positive (Examples) ≠ ⍰;

    Clause = ⍰;

Split Examples (Split Ratio, Examples, Growing Set, Pruning Set)

    Cover = Growing Set

While Negative (Cover) ≠ ⍰ ;

    Clause = Clause ⍰ Find Literal (Clause; Cover)

    Cover = Cover (Clause, Cover)

loop

    NewClause = BestSimplification (Clause, PruningSet)

if Accuracy(NewClause,PruningSet) < Accuracy(Clause,PruningSet)

    exit loop

    Clause = NewClause

if Accuracy(Clause,PruningSet)<=Accuracy(fail,PruningSet)

    exit while

    Theory = Theory ⍰ Clause

Examples = Examples -Cover

    return (Theory)

// end of IREP

//frequent set generation

    for transaction t € Z

    Ck ← Subset(C_k,t)

    for candidates c € Ct

        count[c] =count[c + 1]

    Z_k ← { c € C_k| count[c] >= e}

    k ← k+1

    return Z_k
```

Figure 1: Pseudocode of AIREP algorithm

The basic idea of Incremental Reduced Error Pruning (IREP) is that instead of first growing a complete concept description and pruning it thereafter, each individual clause will be pruned right after it has been generated. This ensures that the algorithm can remove the training examples that are covered by the pruned clause before subsequent clauses are learned thereby preventing these examples from influencing the learning of subsequent clauses.

Figure 1 shows pseudo-code for this algorithm. As usual, the current set of training examples is split into a growing (usually 2/3) and a pruning set (usually 1/3). However, not an entire theory, but only one clause is learned from the growing set. Then, literals are deleted from this clause in a greedy fashion until any further deletion would decrease the accuracy of this clause on the pruning set. Single pruning steps can be performed by submitting a one-clause theory to the same BestSimplification subroutine used in REP or, as in our implementation, one can use a more complex pruning operator that considers every literal in a clause for pruning. The best rule found by repeatedly pruning the original clause is added to the concept description and all covered positive and negative examples are removed from the training growing *and* pruning set. The remaining training instances are then redistributed into a new growing and a new pruning set to ensure that each of the two sets contains the predefined percentage of the remaining examples. From these sets the next clause is learned. When the predictive accuracy of the pruned clause is below the predictive accuracy of the empty clause (i.e., the clause with the body fail), the clause is not added to the concept description and I-REP returns the learned clauses. Thus, the accuracy of the pruned clauses on the pruning set also serves as a stopping criterion. Post-pruning methods are used as pre-pruning heuristics.

In figure 2 the attributes of the dataset are divided into instances and converted into divided attributes. In order to build a rule, IREP uses the following strategy. First the uncovered examples are randomly partitioned into two subsets, a growing set and a pruning set. Next, a rule is grown. It begins with an empty conjunction of conditions, and considers adding to this any condition of the form $Z_n = U_i$, $Z_n <=$ ⍰ or $Z >=$ ⍰ where Z_n is a nominal attribute and u is a legal value for Z_n , or Z_c is a continuous variable and ⍰ is some value for Z_c that occurs in the training data. After growing a rule, the rule is immediately pruned .

After growing a rule, the rule is immediately prunedTo prune a rule, our implementation considers deleting any final sequence of conditions from the rule and chooses the deletion that maximizes the function

$$u(Rule,PrunePos,PruneNeg) = \frac{X + (N - n)}{X + N}$$

where X (respectively N), is the total number of examples in PrunePos ,PruneNeg and p ,n, is the number of examples in PrunePos ,PruneNeg covered by Rule.This process is repeated until no deletion improves the value of u.

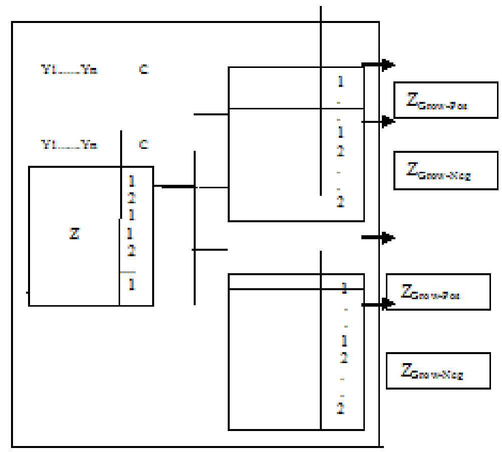

Figure 2. Partitioning of original data set of labelled instances

III. Concept Used

ALARM BASIC CORRELATIONS DISCOVERY ENVIRONMENT ARCHITECTURE (ABCDE)

A. ABCDE architecture overview

Correlation and filtration rules database contains data

about correlations and filtrations to be performed in realtime manner by alarm processing engine. Rules from this database are proposed by Correlation discovery and analysis module. This module can be used for discovery of new potential rules performing data mining algorithm on historical alarm data. It can be used for analysis and evaluation of potential rule candidates also, performing rule execution on sample of historical alarm data. Filtration part of Correlation discovery and analysis module discovers and evaluates potential filter patterns .Alarm data warehouse is a database containing all raw alarm history data as well as correlated alarm history data for a certain time period, predefined by the operator (e.g. 2 years). [1]Alarm data warehouse is starting point for discovery and analysis of typical correlations from alarm historical data, in order to include it in the Correlation and filtration rules database[2].

Correlations are used to determine the root cause of a fault and to filter out redundant alarms (JacquesH. Bellec 2006). A lot of effort have been made researching alarm correlations, resulting in that all alarm systems support advanced filtering mechanisms, Wallin et al. (2009) argues that the problem lies in defining the rules used to filter the alarms. By filtering out all redundant alarms the network operators would only have to handle relevant alarms which would make the network management centermore efficient (Wallin et al. 2009). In a survey from 2009 one representative for a leading telecom operator estimated the use of alarm correlation to 1-2% of all the alarms and the overall attitude of the survey was that the technique is expensive and complex (Wallin & Leijon 2009)[82].

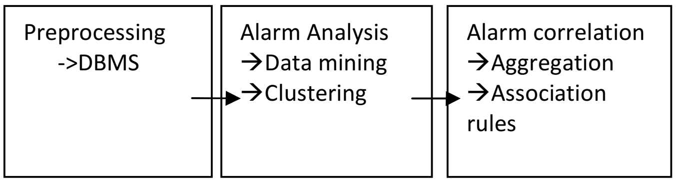

Figure3 Correlation rule generation process.

Incoming network alarms are generated by the telecommunication network. Alarms are consumed and processed by alarm processing engine that performs alarm filtration as well as low and high-level correlation.

Processed alarms are presented to the network operator through alarm surveillance GUI. Alarm processing engine uses correlation and filtration rules stored in database, while incoming alarms are stored into alarm data warehouse.

Logical inventory database containing data about network interconnections can be use for more efficient alarm correlation. Logical inventory data can be used for enhancement of incoming alarm data also, tying relevant inventory information with alarm data (for instance,"friendly" alarm location name). Alarm processing engine is not the focus of this paper since number of commercial tools is able to perform alarms processing functions.

The description of similarity between alarms given by Julisch is based on defined taxonomies. The closest their attributes are within certain taxonomy, the more similar two alarms willLogical inventory data should be obtained from network operator. However, if it is not obtainable, there is proposed technique how to extract logical inventory data from alarm history. It was described in [7], and it is not primary focus of this paper. However, it was denoted on figure 2 through Logical inventory block.
When clusters are generated, the Apriori algorithm is performed. The final result is the number of alarm sequences that occurred frequently in the past. Those sequences are potential high-level correlation rules candidates for future alarm processing. Criteria for acceptation of those candidates can be rule frequency, but also rule can be accepted based on network expert's opinion.

3.1 ALARM DATABASE

To effectively probe for statistics in the database the above mentioned limitations and a simple rule where used when filtering and rebuilding the database. AlarmType is a naming convention and a variable described in the alarm standard document X.733 (a standard in telecommunications). The alarm type variable is used in the uam for alarm name and mapping to the original alarm specifications. If the alarm type could not be found the alarm was deleted from the database.[76,75]

After several years of research on IDS, the variety of results obtained has made the scientific community conclude that further research is needed to fine tune these systems. Large organisation and companies are already setting up different models of IDS from different vendors. Nevertheless, they provide an unmanageable amount of alarms. Inspecting thousands of alarms per day and sensor [1] is not feasible, specially if 99% of them are false positives [2]. Due to this impracticability, during the last four years research on intrusion detection systems has focused on how to handle alarms. The main objectives of these investigation works are: reduce the amount of false alarms, study the cause of these false positives, create a higher level view or scenario of the attacks, and finally provide a coherent response to attacks understanding the relationship between different alarms.

For instance, the manager can decide to launch chips discount for every customer buying 6 beers. The previously mentioned special offer seems to be very logical, based on our daily experience. However, there are numbers of such association rules that cannot be perceived by casual observation. Hence, the manager is forced to analyze the supermarket's transaction data (i.e., customer receipt archive or database) – to examine customer behavior while purchasing products. The result of such analysis is a set of typical association rules describing how often items are purchased together. For instance, rule "Beer _ Chips (80%)" states that four of five customers buying beer are also buying chips [3]. That result can be useful for business decisions related to marketing, pricing and product promotion.

We have considered our alarms as products purchased in a supermarket, and alarm clusters as baskets from a specific customer. Hence we have decided to use the Apriori algorithm in order to find and recognize specific alarm sequences – potential correlation rules for the future [2]. Apriori algorithm itself is described in number of papers such as [3]. The final result of high-level correlations is the creation of a correlation rules database. Rules are structured in an IF-THEN manner. It means that the alarm processing engine will receive incoming alarm stream matching incoming patterns with existing patterns in the correlation rules database. When a pattern is matched, a new alarm is generated containing information about the real network root-cause problem.

IV.Proposed work

The Proposed Algorithm : Pseudocode

• Join Step: Ck is used with Lk-1
• Prune Step: Any (k-1)-itemset that is not frequent cannot be a
subset of a frequent k-itemset
• Pseudo-code: Ck
: Candidate itemset of size k Lk
: frequent itemset of size kL Input: alarm queue (Sij
, Wk)
Output: t frequent alarm sequence set: F_ ALARMm
1. compute C1:={ α | α∈F_ALARM1};
2. m:=1;
3. while Cm≠Φ do
4. begin
5. For all α∈Cm , Search alarm queue Sij
to find support(α, Wk); /*Algorithm 2 */
6. Obtain F_ALARMm={ α∈Cm| support(α, Wk)≥ min_support};
7. Generate Candidate Cm+1 from F_ALARMm; /* Algorithm 3 */
8. m=m+1;
9. end.
10. for all m , output F_ALARMm; {frequent items};
for (k = 1; Lk ;!=∅; k++) do begin
Ck+1
dataset generated from Lk;
for each transaction t in database do
increment the values that are contained in t
Lk+1 // candidates in Ck+1
end
return the resultant rules.
Lk
;

Figure 4:DataAprori algorithm

Alarm correlation algorithm (Algorithm 1) is composed of two main steps. In the first step, according to the minimum support(Min_support), it searches the frequent alarm type sequence from alarm queues and the discovered frequent alarm type sequences constitute the set of frequent alarm type sequences, denoted by F_ALARMm. In the second step, according to the confidence of correlation rule .It generates the alarm correlation rules from F_ALARMm. The association rules algorithm and its measure of association rule S→T, which is defined as confidence(S→T) ,=Support(ST/ Support(S), where S and T correspond to a set of attributes and S and T are disjoint.
The support and confidence of an association rule S->Y are defined asSupport=P[ST] and Confidence=P[ST]/S[T]. The confidence is the conditional probability of Tgiven S. If S and T are independent, then Confidence =P[ST]/P[S]=P[T]. Therefore, if P[T] is high, then the confidence of the rules is high, which will make association rule meaningless. In order to solve the problem.The interestingness measure I=P(ST)/(P(S)×P(T)). The interestingness measure is symmetrical, because the confidence of S->Y is equal to the one of T->S.. A rule holds if and only if the confidence of rule is greater than min_conf.

Input: Frequent alarm sequence set F_ALARMm
Output: output the correlation rules β→(α-β) and confidence |P(α)/P(β)-P(α-β)|
1. for all α F_ALARMm do /* generate correlation rules */
2. for all β α do
3. if|P(α)/P(β)-P(α-β)|≥min_conf then
4. begin
5. generate the rule β→ (α-β) with
6. confidence |P(α)/P(β)-P(α-β)| ;
7. end

V IMPLEMENTATION ASPECTS AND EXPERIMENTAL RESULTS

DataAprori components are developed using C and C++ programming languages, as a parts of complex application.Central application component is executable file thatinvolves different dynamic-linked libraries (dll) in architecture. Every part is implemented as separated dll. It allows upgrade of separated components without disturbing general application structure.

For database access we have used Open Database Connection (ODBC) with all data stored in MS SQL server. For database access we have used standard MFC classes, but all other techniques could be used. The data in experiment 1 are the alarms in GSM Networks, which contain 181 alarm types and 91311 alarm events. The time of alarm events ranges from 1201-03-15-00 to3001-03-79-52. In figure 4 the broken line graph is denoted by win_xy, where x represents the size of additional alarm window i.e.Win_add and y represents the size of frequent alarm window i.e. Win_freq. In figure 4 the Y axis is the number of alarm type sequences and the X axis is Mini_support (using the minimum occurring times)

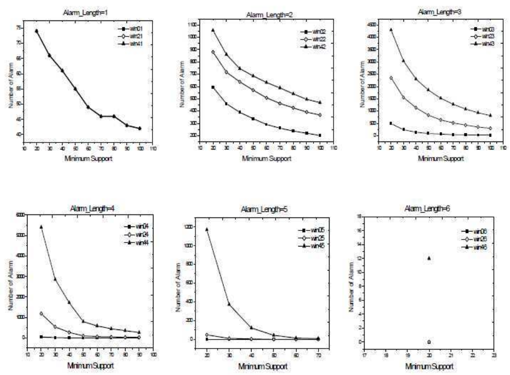

Figure 5: The number of frequent sequences changes in DataAprori.

	Completeness	Time consumption	Support rate	Reduction rate
DataAprori	80-%91%	low	100%	78%
Aprori method	79%	Middle	93%	80%

Figure 6 : Comparison between dataaprori and aprori method.

From Figure 6, we can find that the reduction rate of our method is a little better than aprori method.However aprori method is not able to filter dataset in real time. It can distinguish true alerts and false ones onOur method has low time consumption as compared to the aprori method . Moreover,this method needs a lot of labeled data to build its modeland can not filter alerts in training phrase, while our method does not have these limits. So using our method,security managers can response to attacks more quickly. From above comparison, we believe that our system has better performance than current methods.

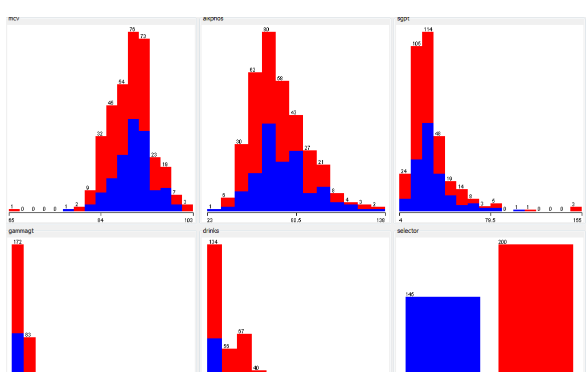

Figure 7 : Comparison of generated rules.

VI. Conclusion

Since the DataAprori algorithm can analyze alarm correlation from alarm database containing noise data, it will generate more alarm sequences, then the number of correlation rules increases. Although the correlation measure can reduce the rules, it still needs people to select the most useful ones from a large number of the rules. Therefore, it is necessary to study how to extract rules more correlated from alarm database containing noise in the future. This number can be reduced if we discover some frequently repeated alarm sequences, and replace it by one alarm. For that purpose, we have used Apriori algorithm, as we discussed in our previous work. However, after sequences are detected, it is necessary to “judge” which sequence is relevant for future and which is not. One of criteria can be frequency of alarm sequence appearing.

Also, some sequences can be very relevant, event if those are not repeated very frequently. DataAprori can be used fordiscovery and statistical processing of alarm sequences,while final decision should be made by human operator.According to our previous and other related works [12],reduction rate at high-level correlations can be rather high,up to 80%. Using test data sample and finding several alarm sequences confirmed by network experts, reduction rate was 15.41 %.

VII. Future work

Further research efforts should be invested into the full implementation of proposed architecture, improving and introducing new data mining techniques for high-level correlations discovery as well as typical patterns that can be used for low-level correlations and filtrations. Fuzzy technique can also be improvised in the proposed DataAprori in future.

ACKNOWLEDGMENT

The authors wish to thank Jamia Hamdard University Library , Labortory for allowing experimentation and research . The author wish to acknowledge ,Prof M Afshar Alam ,Prof Ranjit Biswas and others contributors for developing and contribution to this paper. The author also wants to express gratitude towards the major contributor of the paper Dr Oliver Jukic Virovitica,Dr Kunstic M .Their concept in the field has helped me to scale the existing data mining algorithm.

REFERENCES

I1] Oliver Jukić,Virovitica, Marijan Kunštić, ABCDE – Alarm Basic Correlations Discovery Environment University of Zagreb
Faculty of Electrical Engineering and Computing Zagreb, Republic of Croatia .

[2] Kunštić, M., O. Jukić and M. Bagić, "Definition of formal infrastructure for perception of intelligent agents as problem solvers", Proceedings on International Conference on Software, Telecommunications and Computer Networks, Nikola Rožić and Dinko Begušić (ed.), Split, 2002.

[3] Jukić, O., M. Kunštić, "Network problems frequency detection using Apriori algorithm", Proceedings of the 32rd International Convention MIPRO 2009., Golubić S. et al. (ed.), pp. 77-81, Opatija, Republic of Croatia, 2009.

[4] Goethals, B., "Survey on frequent pattern mining",Department of Computer Science, University of Helsinki, Finland, 2009.

[5] Kowalski, R., Logic for problem solving, North Holland, New York 1979.

[6] Udupa, K.D., TMN – Telecommunications Management Network, McGraw-Hill Telecommunications, New York, 1999.

[7] Jukić, O., M. Kunštić, "Logical inventory database integration into network problems frequency detection process", Proceedings of the 10th International Conference on Telecommunications CONTEL 2009., Podnar Žarko, Ivana; Boris, Vrdoljak (ed.), pp. 361-365, Zagreb, Republic of Croatia, 2009.

[8] Burns, L., J.L.Hellerstein, S.Ma, D.J.Taylor, C.S.Perng, D.A.Robenhorst, "Toward Discovery of Event Correlation Rules", IBM T.J. Watson Research Center, Hawthorne, New York USA

[11] ITU T, Recommendation X.733: Alarm Reporting Function, Geneva 1992.

[12] Garofalakis, M., R. Rastogi, "Data mining meets network management – The Nemesis project", Bell Laboratories, USA, 2001.

[13] Jukić, O., M. Špoljarić, V. Halusek, "Low-level alarm filtration based based on alarm classification", Proceedings of the 51stInternational Symposium ELMAR 2009., Grgić, Mislav et al. (ed.), pp. 143-146, Zadar, Republic of Croatia,2009

[14] Costa, R., N. Cachulo, P. Cortez, "An Intelligent Alarm Management System for Large-Scale Telecommunication Companies", EPIA 2009, L. Seabra Lopes et al. (ed.), pp. 386-399, Berlin 2009

[15] Kunštic, M., O. Jukic and M. Bagic, "Definition of formal infrastructure for perception of intelligent agents as problem solvers", Proceedings on International Conference on Software, Telecommunications and Computer Networks,Nikola Rožic and Dinko Begušic (ed.), Split, 2002.

[16] Jukic, O., M. Kunštic, "Network problems frequency detection using Apriori algorithm", Proceedings of the 32rdInternational Convention MIPRO 2009., Golubic S. et al.(ed.), pp. 77-81, Opatija, Republic of Croatia, 2009.

[16] Goethals, B., "Survey on frequent pattern mining",Department of Computer Science, University of HelsinkiFinland, 2009.

[17] Kowalski, R., Logic for problem solving, North Holland,New York 1979.

[18] Udupa, K.D., TMN – Telecommunications anagement Network, McGraw-Hill Telecommunications, New York,1999.

[19] Jukic, O., M. Kunštic, "Logical inventory database integration into network problems frequency detection process", Proceedings of the 10th International Conference on Telecommunications CONTEL 2009., Podnar Žarko,Ivana; Boris, Vrdoljak (ed.), pp. 361-365, Zagreb, Republicof Croatia, 2009.

[20] Burns, L., J.L.Hellerstein, S.Ma, D.J.Taylor, C.S.Perng,D.A.Robenhorst, "Toward Discovery of Event CorrelationRules", IBM T.J. Watson Research Center, Hawthorne, New York USA

[21] ITU T, Recommendation X.733: Alarm Reporting Function,Geneva 1992.

[22] Garofalakis, M., R. Rastogi, "Data mining meets network management – The Nemesis project", Bell Laboratories,USA, 2001.

[23] Jukic, O., M. Špoljaric, V. Halusek, "Low-level alarm filtration based based on alarm classification", Proceedings of the 51stInternational Symposium ELMAR 2009., Grgic,Mislav et al. (ed.), pp. 143-146, Zadar, Republic of Croatia,2009

[24] Costa, R., N. Cachulo, P. Cortez, "An Intelligent Alarm Management System for Large-Scale telecommunication Companies", EPIA 2009, L. Seabra Lopes et al. (ed.), pp.386-399, Berlin 2009.

[25]R. Uday kiran, P Krishna Reddy, "An Improved Multiple Minimum Support Based Approach to Mine Rare Association Rules", IEEE Symposium on Computational Intelligence and Data Mining , 2009.

[26] Anurag Choubey, Ravindra Patel, J.L. Rana, "A Survey of Efficient Algorithms and New Approach for Fast Discovery of Frequent Itemset for Association Rule Mining (DFIARM)", International Journal of Soft Computing and Engineering (IJSCE), Volume-1, Issue-2, May 2011.

[27] M.S.Danessh, C. Balasubramanian and K. Duraiswamy, "Similarity Data Item Set Approach: An Encoded Temporal Data Base Technique", Journal of Computing, Volume 2, Issue 3, March 2010.

[28] Jianying Hu, Aleksandra Mojsilovic, "High-utility pattern mining: method for discovery of high-utility item sets", Science Direct, The Journal of pattern recognition society, pg 3317 – 3324, 2007.

[29] Jyothi Pillai, O.P.Vyas, "Overview of Item set Utility Mining and its applications", International Journal of Computer Applications, Volume 5- No.11, August 2010.

[30] Fan Lilin, "Research on Classification Mining Method of Frequent Itemset", Journal of Convergence Information Technology Volume 5, Number 8, October 2010.

[31] Laszlo Szathmary, Petko Valtchev, and Amedeo Napoli2, "Finding Minimal Rare Itemsets and Rare Association Rules", Proceedings of the 4th International Conference on Knowledge Science, Engineering and Management pg 16-27,2010.

[32] Michael Hahsler and Bettina Grun and Kurt Hornik and Christian Buchta, "Introduction to a rules: A computational environment for mining association rules and frequent item sets", PhD thesis, March 24, 2009.

[33] J.Shahrabi and R. S. Neyestani, "Discovering Iranians' Shopping Culture by considering Virtual Items Using Data Mining Techniques ", Journal of Applied Sciences Volume-9 Issue-13, pg 2351-2361, 2009.

[34] Tarek F. Gharib, Hamed Nassar, Mohamed Taha, Ajith Abraham, "An efficient algorithm for incremental mining of temporal association rules", Science Direct Data & Knowledge Engineering, Data & Knowledge Engineering , pg 800-815 ,2010.

[35] Neelu Khare ,Neeru Adlakha ,K. R. Pardasani, "An Algorithm for Mining Multidimensional Fuzzy Association Rules, International Journal of Computer Science and Information Security,Vol. 5, No. 1, 2009 .

[36] M. Mart´ınez-Ballesterosa, A. Troncosob, F. Martinez A´ lavers and J. C. Riquelmea, "Mining quantitative association rules based on evolutionary computation and its application to atmospheric pollution", Integrated Computer-Aided Engineering , 227–242,2010.

[37]Tzung-Pei Hong, Chun-Wei Lin, Yu-Lung Wu , "Incrementally fast updated frequent pattern trees", Expert Systems with Applications , pg 2424–2435 , 2008 .

[38] Maja Dimitrijevic, Zita Bosnjak, "Web Usage Association Rule Mining System ", Interdisciplinary Journal of Information, Knowledge, and Management Volume 6, 2011.

[39] Suraj Srivastava, Deepti Gupta, Harsh K Verma, "Comparative Investigations and Performance Evaluation for Multiple-Level Association Rules Mining Algorithm", International Journal of Computer Applications Volume 4– No.10, August 2010.

[40] Pratima Gautam, Neelu Khare K. R. Pardasani, "A model for mining multilevel fuzzy association rule in database ", Journal of computing, Volume 2, Issue 1, January 2010.

[41] S. Lofti, M.H. Sadreddini, "Mining Fuzzy Association Rules Using Mutual Information", Proceedings of the International MultiConference of Engineers and Computer Scientists , pg 18 - 20, March 2009, Hong Kong.

[42] Fang Li, Chengyao Li, and Yangge Tian, "Applying Association Rule Analysis in Bibliometric Analysis", Proceedings of the Second Symposium International Computer Science and Computational Technology China, pg 431-434, Dec. 2009.

[43] Preetham Kumar, Ananthanarayana V. S, " Discovery of Multi Dimensional Quantitative Closed Association Rules by Attributes Range Method", Proceedings of the International MultiConference of Engineers and Computer Scientists Vol-I,pg 19-21, March 2008, Hong Kong.

[44] Rakhi Garg, P. K. Mishra, "Exploiting Parallelism in Association Rule Mining", International Journal of Advancements in Technology Vol No- 2 ,April 2011.

[45] A.Meenakshi, K.Alagarsamy, "A Novelty Approach for Finding Frequent Itemsets in Horizontal and Vertical Layout-HVCFPMINETREE", International Journal of Computer Applications Volume 10– No.5, November 2010.

[46] C.S.Kanimozhi Selvi and A.Tamilarasi, "An Automated Association Rule Mining Technique with Cumulative Support Thresholds", Int. J. Open Problems in Compt. Math, Vol. 2, No. 3, September 2009.

[47] Kahayan Lal, N.C.Mahanti, "Mining Association Rules in Large Database by Implementing Pipelining Technique in Partition Algorithm", International Journal of Computer Applications Volume 2 – No.4, June 2010.

[48] E. Chandra, K. Nandhini, "Knowledge Mining from Student Dat a ", European Journal of Scientific Research No-1, pg156-163, 2010.

[49] Przemyslaw Kazienko, "Mining Indirect Association rules for web recommendation ", International Journal of Applied Mathematical Computing Science, Vol. 19, No. 1, 165–186,2009.

[50] Fadi Thabtah, "Pruning Techniques in Associative Classification: Survey and Comparison ", Journal of Digital Information Management, Volume 4

[51] J S. Park and M.S. Chen and PS. Yu, "An effective hash-based algorithm for mining association rules", International Conference on Management of Data", pg 175-186, May 1995.

[52] Floriana Esposito, Donato Malebra, Giovanni Semeraro, Valentina Tamma,"The effects of pruning methods on the predictive accuracy of induced decision trees." ,Applied Stochastic. Models Business and Industry, pg 277-299, 1999.

[53] Niblett T, Bratko I, "Learning decision rules in noisy domains. Proceedings of Expert Systems "Cambridge University, 1986.

[54] K. P. Bennett, "Decision Tree Construction Via Linear rogramming." Proceedings of the 4th Midwest Artificial Intelligence and Cognitive Science Society, pp. 97-101, 1992.

[55]Dataset source http:// software.seg.org /datasets/2D /Hess_VTI /, ftp.cs.wisc.educdmath-prog/cpo-dataset/machine-learn/ WDBC/.

[56] Sapna Jain , Tamanna Siddiqui, M Afshar Alam, A I R E P: A novel scaled multidimensional quantitative rules generation approach.

[57] Oliver Jukic,, Marijan Kunštic, Virovitica College Virovitica, Republic of Croatia.

[58] http:/ /www .mipro.hr /News /NagrađeniradoviMIPRO2010/tabid/134/language/en-US/Default.aspx

[59] http://www.mipro.hr/MIPRO2010.CTI/ELink.aspx

[60]http://www.slideshare.net/zafarjcp/data-mining-association-rules-basics.

[61]http://www.slideshare.net/zafarjcp/data-mining-association-rules-basics.

[62] http://www.kdnuggets.com/data_mining_course/x3-algorithms-decision-trees-intro.html

[63] http://portal.acm.org/citation.cfm?doid=170035.170072

[64] http://lacam.di.uniba.it:8000/people/semeraro.htm

[65] http://www.krpardasani.com/list1.htm

[66] http://www.authorstream.com/ Presentation/ Octavio-12948-Lecture2-Entertainment-ppt-powerpoint/.

[67]http:// edu.eap.gr/pli/ pli10/info/CVs/ Kotsiantis_CV.htm.

[68] http://hyoka.ofc.kyushu-u.ac.jp/search/ details/ K000197/thesisList.html

[69] http://kkaneko.com/kaneko/index-j.html

[70] https://weblogs.sdn.sap.com/pub/wlg/13598

[71] http://dl.acm.org/citation.cfm?id=170072

[72] http://perweb. firat.edu.tr /default.asp? content= personelgoster.asp&uid=M-A-0284

[73] http://sites.google.com/site/ijcsis/vol-8-no-6-sep-2010

[74] http://academic.research .microsoft.com/Paper /378418

[75] INTRUSION DETECTION ALARM CORRELATION: A SURVEY Urko Zurutuza, Roberto Uribeetxeberria.

[76] Jesper Lindberg,Daniel Nilsson,VALIDATING A UNIFYING ALARM MODEL.